Payoff

As the role of data in an organization expands and data becomes increasingly related to profitability, the impact of shared data access must be measured against the total business operating environment. This article discusses the data resource management function and its responsibilities and accountabilities related to distributed processing environments.

Introduction

It is difficult to overstate the role that data plays in an organization. Data provides the basis for management control and decision making, measures the degree to which customers' expectations and suppliers' commitments have been attained, and records performance for shareholders and government agencies. The role of data is expanding as organizations extend their original business to include the sale of data management services. For example, airline reservation systems serve travel agencies, governments sell data from vehicle registrations to auto supply organizations, telephone companies provide billing services for 900 numbers, and banks provide billing and payroll services.

Organizations recognize the value of shared data access. Increasingly, business managers need to understand the role of the data resource is overall profitability and strategic planning. This article explains the different functions of Data Resource Management and, more important, their significance in a business environment that increasingly demands distributed processing and requires corporate accountability for data management.

Data Management and Corporate Accountability

The data management function encompasses such varied areas as:

- Policy, strategy, standards, and guidelines.
- Data architecture (including use of a dictionary and repository).
- Physical data base design.
- Security issues (including user access and direct external interfaces).
- Auditing and integrity.
- Provisioning of data to a complex computer environment.
- Provisioning of data to complex application systems.
- Integration and testing.
The term provisioning refers to the movement, transfer, or extraction of data from one system, machine, or data center to another.

These functions have overlapping dependencies. For example, policies and strategies, together with their supporting methodologies, tools, and standards, apply to all other Data Resource Management functions. Security and audit functions apply to the core data resource management responsibilities of managing the data architecture, data base design, integration, testing, and application use.

Policy, Strategy, and Standards

Data is a corporate resource—that is, just as no single business unit owns furniture or supplies, no single unit owns data. Each unit has a responsibility for collecting and protecting data in the same way that it would any other asset of the organization. By giving customers and suppliers direct access to a data base, an organization provides the equivalent of direct access to its product warehouses. To protect its exposure and avoid vulnerability, an organization must establish policies, strategies, standards, and guidelines relating to the sharing, security, integrity, redundancy and replication of data as well as access to it.

Strategies relating to the use of data must address such issues as the equitable treatment of all customers and suppliers and avoidance of restraint of trade. In addition, the organization must protect the privacy and confidentiality of customer information and trade secrets. Legislation may also affect the transfer of data across international and regional boundaries.

The key responsibility of the data policy group is to establish a corporate direction for shared data before technological solutions are implemented that jeopardize the organization's reputation and market position. In addition, it must set standards and guidelines for developing information systems as well as for keeping pace with changing technology, legislation, and business opportunities.

The Data Architecture

The key responsibilities of the data architecture group are:

- Developing the business and data models.
- Maintaining the repository of diagrams, definitions, access, and security requirements.
- Ensuring that the data base designs accurately reflect the business entities and policies.

An integrated data architecture defines the business objects and serves as the blueprint for a shared data base design. A data model describes business entities and the policies relating to them and records semantic meanings and properties of each entity. Although the data model forms the basis of the data base design, it remains independent of specific hardware and software technologies. Because it reflects the semantic meaning of the business entities, the data model can be used as a communications vehicle to train employees, to convey systems and data base requirements to management, and to optimize business operations.

Each new business application extends the integrated data architecture. The data model consists of a diagram showing all the entities and their relationships together with supporting entity and attribute definitions. Each entity has a custodian responsible for
defining its meaning, specifying the edit and audit rules for its attributes, and defining its access and security requirements. For example, human resource application uses entities describing Employee, Skill, and Work Location. The construction plan application uses entities describing Employee, Skill, Equipment, and Inventory. Common to both applications are the Employee and Skill entities, and these are integration points between applications. The materials management application is connected to the integrated model through the Inventory entity.

Physical Data Base Design

Physical data base designs represent the automated portion of the business and data architecture. Ideally, there would be one-to-one mapping between the entities on the data model and the tables built in a relational DBMS. For nonrelational DBMSs, an entity would correspond to a segment on a hierarchical data base or a record type in a file.

Designs Supporting Objects.

Maintaining the consistency between business objects and data base objects supports object-oriented application design techniques. Modifying application as the business changes then becomes a relatively straightforward process. Individual tables, segments, and record types can be added or deleted with minimal impact on the balance of the data used by the applications.

The Independence of the data architecture from the physical data base design also allows organization to support complex operating environments. Whenever the data needs to be distributed either geographically or between different types of machines (possibly from different vendors), or a combination of both, the physical data base design group administers the data distribution through mapping to the data architecture. The machine types and communication channels can be modified independently of changes to the business operations. The growing demand for distributed processing reinforces the benefits of implementing physical data base designs that correlate to the business objects.

The physical data base design group generates the source code for the data base objects, reflecting the constraints of the hardware and software in the operating environment. In addition, when developing the integrated data base design, the designers balance the index requirements, data volumes, and transaction frequencies to establish the user community service levels, free space allowances, and data set sizes. Changes and enhancements to the physical design must balance the requirements of the entire user community. Several users may be affected when columns or rows are added to a table.

For example, the addition of a series of products in a new product line may require changes to the pricing programs in a customer billing system as well as to the edit programs in the product specification system. The physical data base design must be maintained in a sharable repository to manage change control, to allow application enhancement planning by system designers, and to facilitate hardware and software upgrades and environment planning by the computer facilities group.

Security of User Access and Direct External Interfaces

Security issues relating to data and applications are made more complex because of direct user access facilities, and decentralized computer sites add further to the complexity of security management. Privacy legislation also affects the amount and type of information that can be provided or withheld.
Developers working to create full-function distributed DBMSs recognize that maintaining a secure environment and meeting privacy requirements are priorities. Direct customer and supplier access to an organization's data forces organizations to streamline operational procedures and at the same time build safeguards to control access.

For example, customers may access their own records but not those of any other customer or those containing any data that is not customer related. The ability to sell data access—as shown by airline reservations systems that provide services to tens of thousands of travel agents, or banks that provide account updates to retail stores through debit cards—fuels the growing demand organizations face for authorizing direct external interfaces. In addition, Electronic Data Interchange standards exist to facilitate company-to-company information flow.

With these developments, an organization must have the policies, guidelines, and procedures in place to protect its data against illegal access. Procedures relating to passwords for terminals, users, data sets, and encryption routines must be developed and enforced. Attempts at illegal access must be monitored, and violations identified. The data must also be protected from illegal access in the form of vandalism—that is, computer virus attacks. Recovery procedures can go a long way to reducing the impact of a breach of security, an act of God, or vandalism. All security measures must also extend to off-site storage facilities.

Specifically, criteria must be defined that specify:

- Who possesses authorized access to the data.
- When the data is to be updated.
- When the data is to be purged or destroyed.
- Who is authorized to borrow the data.

The same issues need to be addressed for copies of data sets made for distributed sites. Recording logs for policies, procedures, authorizations, and accesses should be managed both at the local site and organizationwide.

**Audit and Integrity of Data Bases**

The audit group verifies that the content of the data bases and libraries matches the application design and follows the policies and standards that have been set. For example, the ability of a clerk to review receivables or payables before distributing these documents has been eliminated by automatic Electronic Document Interchange. As a result, the organization's reputation is potentially in jeopardy because its shortcomings with regard to data integrity are immediately visible.

Error correction is no longer a simple task of adjusting an entry or rerunning a batch job. Detecting the program errors in customer-accessed transactions requires fallback procedures that contain the extent of the damage. In addition to extensive initial testing of both the application functions and security procedure, organizations must conduct ongoing audit and integrity checks of data base contents.

Auditing data elements for correct use and interpretation is becoming increasingly important. Although serially numbered identifiers are popular for such items as invoices and receipts, they are difficult to edit. For example, it may be possible for the customer or supplier to transpose digits or enter a random number and yet still manage to key a valid
identifier. Checks for reasonableness on data fields that cannot have specific edits are also increasingly important. Automated audits and verifications provide early detection of problems.

An inventory of system objects provides management with the vehicle to ensure that the systems contain the complete set of appropriate objects. Some examples include:

- Operations failure logs.
- User catalogues.
- Program libraries.
- Data set catalogues.

By failing to maintain a precise inventory of system objects, numerous organizations lack the source code necessary to change a program. A specific inventory allows an organization to:

- Meet security requirements.
- Confirm accurate implementation of system designs.
- Provide volume information for computer capacity planning.
- Calculate use data for planning technology changes.
- Identify shared data when making application changes.

**Provisioning Data to Complex Computer Environments**

The provisioning group develops and maintains procedures to manage extracts or copies of data to be used by various computers. The advent of complex operating environments has changed the meaning of shared data. In addition to sharing data across applications, organizations must now also distribute data and programs across decentralized computer sites and diverse hardware and software combinations. This data distribution must comply with authorization and security standards. Copy, change, and security controls are partially enforced at both the user level and the workstation level. Automated procedures to manage the provisioning requirements increase in complexity with the number of sites and machines. Data collection from decentralized sites to a central facility or provisioning data from a central environment is more manageable than transfers of data between workstations. Managing data transfers on diskettes between hundreds of microcomputers can be impossible in some cases.

Management of data copies includes requirements for disaster planning (e.g., off-site storage and archival records). Requirements to retain data must also take into account the program code that created the data. For example, data retained for seven years can be meaningless if the programs that capture and interpret the data are not also saved. Similarly, changing from a batch to an online environment may affect data already archived. Application development groups must consider the cost to the business of ignoring the conversion of historical data to new formats.
The provisioning group must document both the data and the programs used to create archival records—including disaster planning records—and extracts for each computer site. In addition, it must monitor the data and programs being archived to ensure that they meet the security and audit requirements.

**Provisioning Data to Complex Application Systems**

In situations in which applications do not share common data bases, the provisioning group must manage the data transfers through bridging and migration programs. The organization's reputation and its ability to fulfill customer needs are affected by the accuracy of its data. In addition, the time required by employees to reconcile differing business reports affects the organization's profitability. Each time the data is moved or transformed before being used by a different application, the opportunity for error increases.

Complex transfers of data from one application to another or others occur in several situations, such as:

- **Mixed batch and online access.** These transfers involve at least one application accessing the data with online transactions; other applications using the data operate in batch mode.

- **Incompatible mixes of hardware and software.** These transactions involve at least one application using a combination of hardware and software that is incompatible with that operating other applications using the same data.

- **One-to-many transfers.** These occur when one application captures all the data that must then be transferred to other applications for referencing and processing.

- **Many-to-many transfers.** These are data transfers in which many applications capture the data that must be transferred to other applications for processing.

The difficulty of maintaining data integrity is compounded whenever data is captured at different frequencies, under different edit criteria, and using different technologies. The provisioning group must work with the application developers to determine the most efficient and reliable way to merge disparate data stores and maintain reference files.

When converting systems to use a shared data architecture, the provisioning group and application developers should follow a logical build sequence based on the business life cycle. Even so, the period of transition between implementation of the first application supporting shared data and migration of the balance of the applications sharing that data can be significant.

Implementing a single data capture source may require interim back-bridging between the enhanced applications and the down-stream users of that data until all the old data capturing programs have been phased out. For example, for six inventory shipment applications capturing receiving data to be replaced by a single receiving capture routine, the conversion should be phased over two years. Until the last application has been upgraded, the data captured by the new routine will be converted or bridged back to the old format. The complexities vary depending on the interfaces and the number of systems involved.
Integration and Testing

Because of evolving computer environments and shared data architectures, today's requirements for application testing are insufficient to meet requirements for data integrity and security. It is no longer acceptable for a set of interlocking applications to be down for several days while the implementation bugs are resolved in the latest application or enhancement. It is equally unacceptable for systems supporting interactive customer and supplier transactions to be unavailable.

Automating a new business application or enhancing an existing one requires that the impact of its data and processing be measured against the total operating environment. Important aspects of integration and testing include:

- Extending the testing system interfaces for compatible data based on edits, consistent coding structures, and equal field lengths and types to include sets of data. For example, organizations should assess the impact of adding dealers to a customer database.

- Reviewing hardware requirements for data types and indexing capabilities. For example, the organization should know whether the software supports null values.

- Matching the service hours (i.e., with regard to time zones and daily available hours) between the distributed and central sites.

- Allowing adequate windows for batch processing and data provisioning.

- Ensuring that online updates and batch processing do not lose or overlook records during migration and bridging processes.

- Maintaining performance levels for interlocked applications across the extended user bases.

Conclusion

Data represents a permanent and stable resource for an organization. It does not flow from one application to another, nor does it change state as it is used by different applications. For example, data that enters an organization as a Customer Order does not become a Purchase Order against a Vendor, nor does it become a Customer Shipment. Instead, the organization maintains the three separate entities for which it keeps data and measures performance. The data architecture represents defined business objects, and policies specify the relationships between them and the processes that act upon them. Physical data base designs implement the objects and relationships, and program code enforces the relationships and applies business transactions. Security, audit, and copy management ensure the data's safety and integrity and protect its users' privacy rights. Managing migration, bridging, integration, and testing ensures that the data maintains its consistency during the transition phases of implementing the shared and distributed data environment.

The requirement to provide accurate information about the organization's operations to shareholders, government agencies, and customers requires corporate accountability for data management. The data management task carries the same responsibilities and accountabilities as are associated with any corporate asset directly affecting profitability. Although organizations may operate individual units as profit centers, it is overall
performance that counts. Organizations that recognize the intrinsic value of data and make it available as a product to customers have positioned themselves so as to improve their profitability significantly.

Bibliography


Author Biographies

Barbara Grant

Barbara Grant is manager of diversion analysis and support for the Waste Reduction Office of the Ontario Ministry of the Environment, where she is responsible for the province's solid waste management information systems. She has 15 years' experience in information resource management and has developed repository documentation standards for organizations in the United States and Canada.